CLAIMS

What is claimed is:

1. A method for minimizing the cycle time when compiling a program in a computer system, the program including a plurality of directories and each of the directories including a code file; the method comprises the steps of:

(a) providing a master array of directories of the program, wherein the master array lists the dependencies of the directories,

(b) providing a code change to the program to provide an updated program;

(c) providing associated dependency changes to the master array to provide an updated master array; and

(d) compiling the updated program utilizing the updated master array wherein the code files of the directories are compiled in an ordered manner based upon the dependencies of the plurality of directories.

2. The method of claim 1 wherein a scheduler is utilized to compile the updated program, wherein the scheduler receives the dependency changes and a list of processors from a processor array.

3. The method of claim 1 wherein the associated dependencies changes are provided (c) via a directory update mechanism.

4. The method of claim 3 wherein the providing an update mechanism step (c) further comprises the steps of:

(c1) providing an array of dependency changes; and

(c2) merging the dependency changes array with a master array of changes.

5. The method of claim 4 wherein the merging step (c2) comprises the steps of:

(c21) obtaining a dependency change from the dependency changes array;

(c22) determining whether the dependency change is in a directory in the master array;

(c23) updating the directory in the master array of the dependency change in a directory of the master array;

(c24) adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array;

(c25) determining if there is another dependency change in the dependency changes array after either step (c23) or step (c24); and

(c26) repeating steps (c21) – (c25) until all dependency changes have been obtained from the dependency change array.

6. A system for minimizing the cycle time when compiling a program in a computer system, the program including a plurality of directories and each of the directories including a code file, the method comprises the steps of

(a) providing a master array of directories of the program, wherein the master array lists the dependencies of the directories,

(b) providing a code change to the program to provide an updated program;

(c) providing associated dependencies changes to the master array to provide an updated master array; and

(d) compiling the updated program utilizing of the updated master array wherein the code files of the directories are compiled in an ordered manner based upon the dependencies of the plurality of directories.

7. The systemof claim 6 wherein a scheduler is utilized to compile the updated program, wherein the scheduler receives the dependency changes and a list of processors from a processor array.

8. The system of claim 6 wherein the associated dependencies changes are provided (c) via a directory update mechanism.

9. The system of claim 8 wherein the providing an update mechanism step (c) further comprises the steps of:

(c1) providing an array of dependency changes; and

(c2) merging the dependency changes array with a master array of changes.

10. The system of claim 9 wherein the merging step (c2) comprises the steps of:

(c21) obtaining a dependency change from the dependency changes array;

(c22) determining whether the dependency change is in a directory in the master array;

(c23) updating the directory in the master array of the dependency change in a directory of the master array;

(c24) adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array;

(c25) determining if there is another dependency change in the dependency changes array after either step (c23) or step (c24); and

(c26) repeating steps (c21) – (c25) until all dependency changes have been obtained from the dependency change array.

11. A computer readable medium for minimizing the cycle time when compiling a program in a computer system, the program including a plurality of directories and each of the directories including a code file, the method comprises the steps of

(a) providing a master array of directories of the program, wherein the master array lists the dependencies of the directories,

(b) providing a code change to the program to provide an updated program;

(c) providing associated dependencies changes to the master array to provide an updated master array; and

(d) compiling the updated program utilizing the updated master array wherein the code files of the directories are compiled in an ordered manner based upon the dependencies of the plurality of directories.

12. The computer readable medium of claim 11 wherein a scheduler is utilized to compile the updated program, wherein the scheduler receives the dependency changes and a list of processors from a processor array.

13. The computer readable medium of claim 11 wherein the associated dependencies changes are provided (c) via a directory update mechanism.

14. The computer readable medium of claim 13 wherein the providing an update mechanism step (c) further comprises the steps of:

(c1) providing an array of dependency changes; and

(c2) merging the dependency changes array with a master array of changes.

15. The computer readable medium of claim 14 wherein the merging step (c2) comprises the steps of:

(c21) obtaining a dependency change from the dependency changes array;

(c22) determining whether the dependency change is in a directory in the master array;

(c23) updating the directory in the master array of the dependency change in a directory of the master array;

(c24) adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array;

(c25) determining if there is another dependency change in the dependency changes array after either step (c23) or step (c24); and

(c26) repeating steps (c21) – (c25) until all dependency changes have been obtained from the dependency change array.